Food and Drug Administration

System Development Life Cycle

December 1, 2003

Version 1.0

Table of Contents

1.0 INTRODUCTION

The Systems Development Life Cycle (SDLC) is the mechanism to assure that systems under development meet established requirements and support the Food and Drug Administration (FDA) mission functions. It provides a structured approach to managing information systems projects, beginning with the planning processes and ending with the Implementation processes.

The primary audiences for this guidance are the systems developers, project managers, program/account analysts and system owners/users responsible for defining and delivering FDA systems, their staff, and their support contractors. Specific roles and responsibilities are described throughout each life cycle phase. The SDLC is not a substitute for information management skills or experience. The required skill combination will vary according to the project. This guide was developed to disseminate proven practices to system developers, project managers, technical managers and system owners/users throughout the FDA.

The SDLC artifacts are used in the Capital Planning and Investment Process. The relationships are illustrated in Figure 1 below. This document covers in detail the Planning through Implementation Phases. The SDLC Phases grouped into *Select* and *Evaluate* are covered by other controlling guidance. Typical activities are summarized in this document.

The SDLC emphasizes decision processes that influence system cost and usefulness. These decisions must be based on full consideration of business processes, functional requirements, and economic and technical feasibility in order to produce an effective system. The primary objectives of any SDLC are to deliver quality systems that: 1) meet or exceed customer expectations when promised and within cost estimates, 2) work effectively and efficiently within the current and planned information technology infrastructure, and 3) are affordable to maintain and cost-effective to enhance. This FDA SDLC establishes a logical order of events and a common language for conducting system development that is controlled, measured, documented, and ultimately improved.

One methodology does not fit all sizes and types of system development efforts. Therefore, the FDA SDLC methodology provides for a full sequential SDLC work pattern and for alternative SDLC work patterns. Components involved in smaller projects could use these alternative SDLC work patterns, where the documentation is shortened and even combined. It also provides a work pattern to accommodate the acquisition and implementation of commercial-off-the-shelf (COTS) products. The approach shown here allows the use of incremental releases of software elements. A cyclical development and delivery approach, where the desire is to facilitate early delivery of business services and to refine/correct errors in requirements and design through effective user-feedback, is also acceptable.

Note that this SDLC is not meant to be a comprehensive, one-volume work. Additional policy and process documents will detail the parties, actions, responsibilities, and other elements

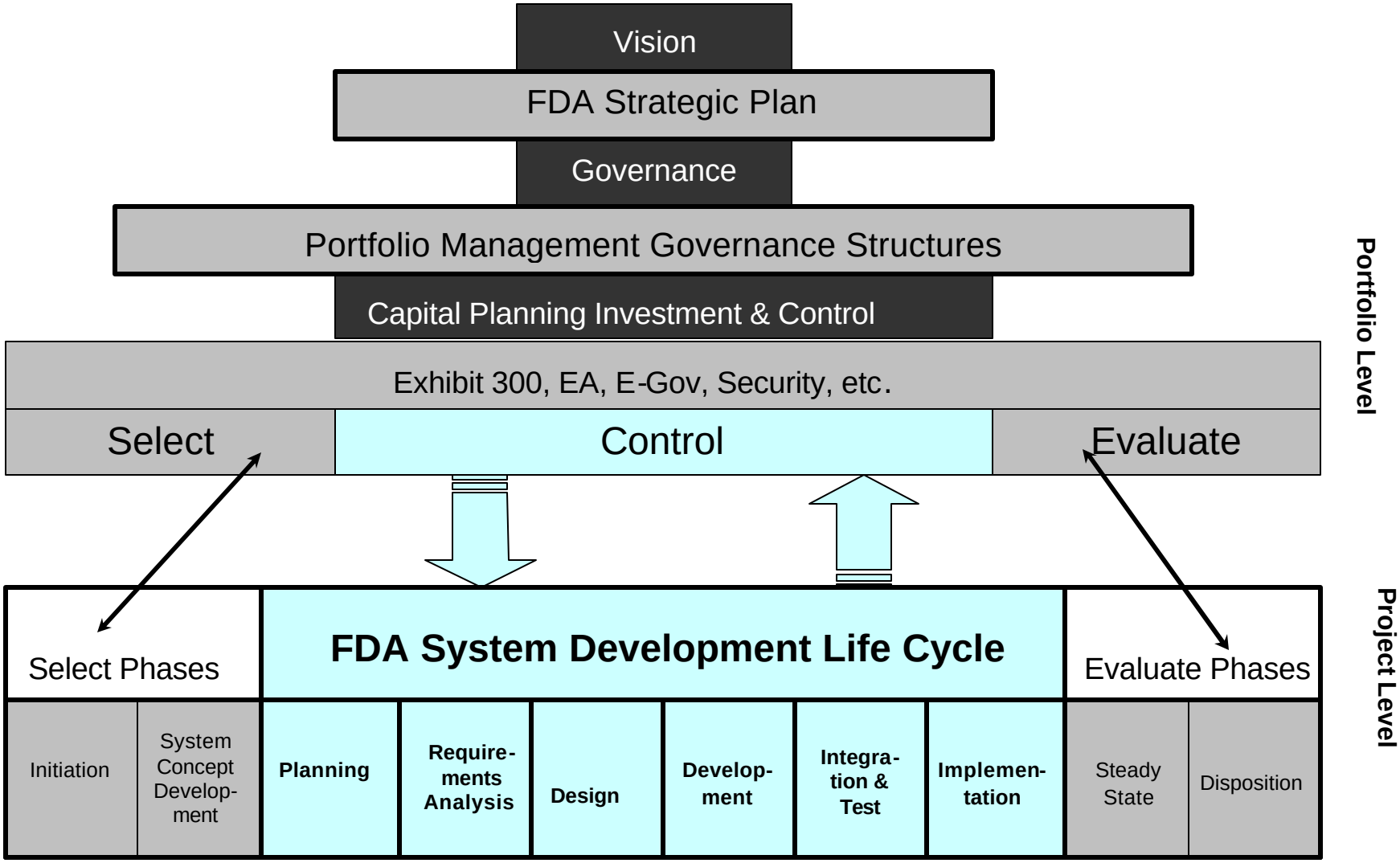


Figure 1: The SDLC's Place in the Capital Planning and Investment Process

associated with the application of the SDLC, including project sizing and the use of pilot projects.[1]

The specific benefits include the following:

- Common system project language to facilitate communication among projects and with business units and management.
- Reduced risk of project failure by encouraging the use of industry best practices.
- Consideration of system and data requirements throughout the entire life of the system.
- Early identification of technical and management issues.
- Disclosure of all life-cycle costs to guide business decisions.
- Fostering realistic expectations of what the systems will and will not provide.
- Information to better balance programmatic, technical, management, and cost aspects of proposed system development or modification.
- Encouragement of periodic evaluations to identify systems that are no longer effective.
- Information that supports effective resource management and budget planning.
- Consideration of meeting current and future business requirements.
- Identifying detailed project schedules

[1] Other documents referenced here include the Office of Management and Budget's Circular No. A.11 "Planning, Budgeting, Acquisition, And Management Of Capital Assets," and OMB's "Guidance for Implementing the Privacy Provisions of the E-Government Act of 200." Procedure documents detailing the application of these and other sources are forthcoming.

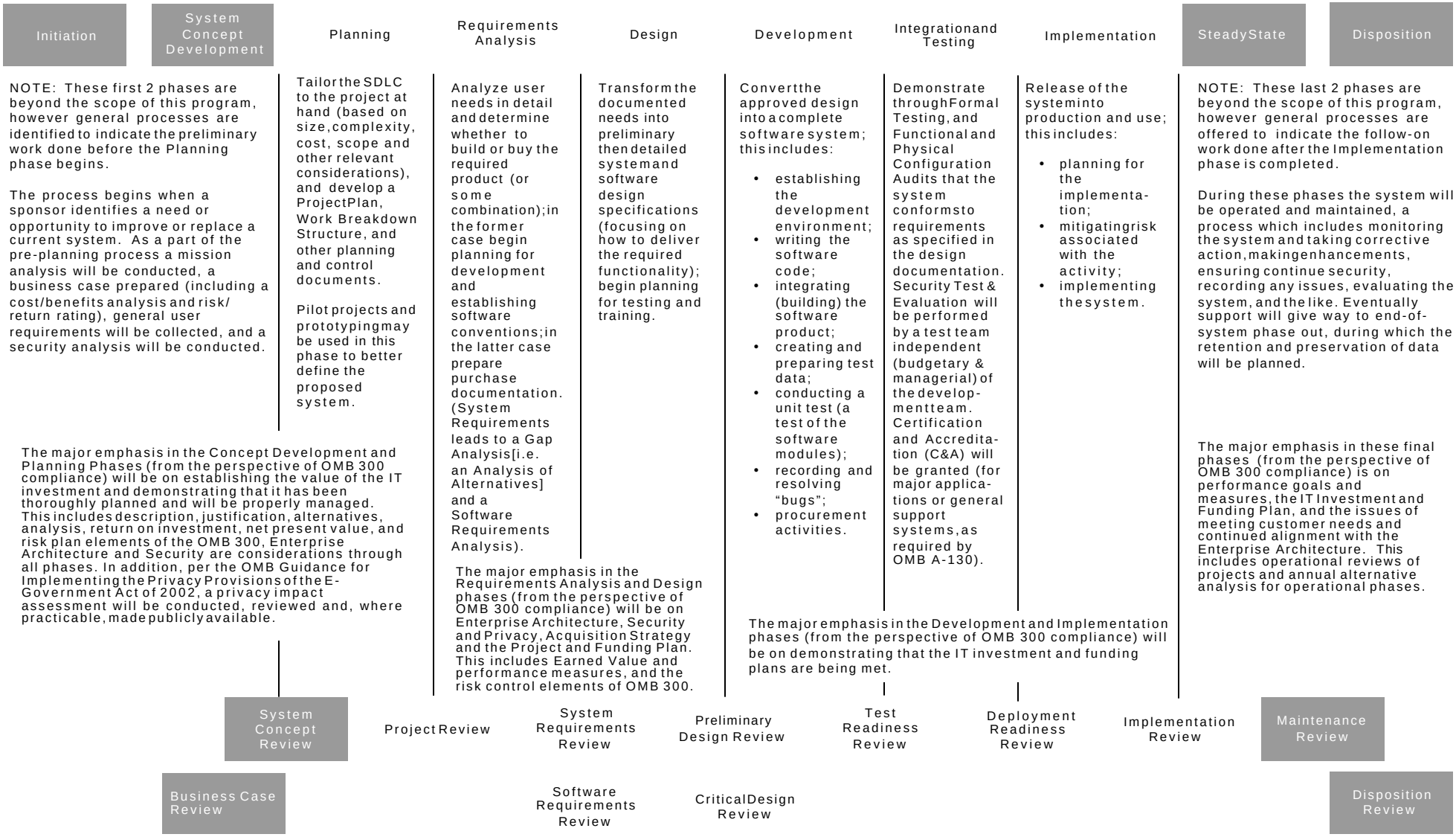


Figure 2: SDLC Phase Overview

2.0 SDLC PHASES

The life cycle of systems in SDLC methodologies are divided into phases. A ten-phase development life cycle is illustrated in Figure 2, below. The six central phases are described in detail in the following sections. Presentation of the first two phases of the SDLC, *Initiation* and *Systems Concept Development*, are beyond the scope of *this* document, as are a detailed presentation of the two concluding phases – *Steady State* and *Disposition* – however a brief overview of their purpose is provided. The phases not covered in detail here are addressed by other guidance documents.

During SDLC phases defined work products and documents are created, reviewed, refined, and approved. The identification of a document in a specific phase is not meant to indicate that the document/artifact is completed during that stage but that work is typically begun on the document/artifact during that stage. The SDLC may be tailored within a Center/Organization to accommodate the unique aspects of a project as long as the resulting approach remains consistent with the primary objective: to deliver a quality system. SDLC phases may overlap and projects can follow an evolutionary development strategy that provides for incremental delivery of products and/or subsystems.[2] Note that not every project will require all phases, all deliverables identified here, or all potential controls.[3]

Each year, the FDA submits a budget to DHHS and OMB. The FDA budget is rolled up with the budgets of other Departments to form the President's Budget, which is submitted to Congress for approval. OMB business case Exhibit 300's are an important element of the budget package that is submitted to DHHS and OMB for Executive branch budget review. Exhibit 300 business cases describe major capital projects including IT projects and the proposed funding needed to accomplish these projects. The SDLC guides development of FDA IT systems that are presented to OMB as business cases. Many products and artifacts of the SDLC processes provide, or are direct information inputs to the Exhibit 300s.

The budget cycle is an annual process with periodic reporting. The SDLC follows the lifecycle of the project, whether that spans months or years. The development of an OMB business case is an iterative process that is linked to the SDLC through information generation and sharing. Exhibit 300s are refined in the FDA budget process from version to version and year to year using SDLC products and artifacts such as alternatives analysis, risk assessment, cost estimates and return on investment analysis. Exhibit 300s can be updated continuously with this and other information from the SDLC process.

Throughout the SDLC the Project Management function is the review and control mechanism for the actions that take place.

[2]Subsystem is understood here as a system that is part of some larger system, but which may itself have inputs, processes, and outputs; multiple sub-systems are generally required to satisfy the overall specification for the system. A system is an integrated collection of subsystems and functions that accomplish an overall goal.

[3]Note that the word "project" here is understood as a temporary operation established to produce some specific expectations; the results of this are then used to change or supplement the normal business operations.

3.0 SELECT PHASES

3.1 Initiation Phase

The Initiation Phase begins when a business sponsor identifies a need or opportunity to improve or replace a current system, or new system requirements are identified. (Note that *system* indicates that software or a combination of software and hardware may be required). The purpose of the Initiation Phase is to:

- Identify and define an opportunity to improve business operations, or support required added functionality for the center/organization.
- Identify significant assumptions and constraints on solutions to that need.
- Recommend the exploration of alternative concepts and methods to satisfy the need including questioning the need for technology.
- For those projects, which require it, begin conducting an Analysis of Alternatives.
- Ensure executive sponsorship.
- Prepare initial documentation to begin control of the project (this includes elements defining the scope, budget, and security concerns).
- Begin the Security and Privacy Assessment process.
- Verify that the project is in alignment with the approved Enterprise Architecture[4] and Capital Planning and Investment Control [CPIC] process.[5]
- Conduct a Sensitivity and Criticality Assessment (described below), which will form the baseline of the security process continuing through each life-cycle phase.
- Conduct a privacy impact assessment, properly reviewed and (if applicable) make this publicly available.[6]
- Ensure that acquisition strategy is addressed in the Business Case or in a separate document.

During this phase the business focus will be on illustrating strategic alignment, analyzing "as-is" processes (including any architectural impact), and measuring current ("as-is") program performance (which will provide a baseline against which to measure alternatives).

During this phase, and all following phases, the Data Council will focus on certifying data standards compliance with accepted FDA data standards, and the effective integration of business and data element architecture.[7]

[4]Reference http://intranet.fda.gov/oirm/ea

[5]This is defined, per the Office of Management and Budget, as a process to structure budget formulation and execution and to ensure that investments consistently support the strategic goals of the Agency.

[6]Per the "OMB Guidance for Implementing the Privacy Provisions of the E-Government Act of 2002" and FDA CIO Memorandum 'Privacy Impact Assessment (PIA), October 24 2003.

[7]The Data Council is chartered with developing information management capabilities and policies.

3.2 System Concept Development Phase

The System Concept Development Phase begins after a business need or opportunity is validated by the Center/Organization Program Leadership and the Center/Organization IT Director. The purpose of the System Concept Development Phase is to:

- Identify basic functional and data requirements to satisfy the business need.
- Establish system boundaries; identify goals, objectives, critical success factors, and performance measures.
- Evaluate costs and benefits of alternative approaches to satisfy the basic functional requirements.
- Verify that the project is in alignment with the approved Enterprise Architecture.
- Begin assessing project risks.
- Identify and initiate risk mitigation actions.
- Continue the security process begun in the Initiation Phase by identifying required and discretionary security controls.
- Complete the Analysis of Alternatives begun in the previous phase.

This phase explores potential technical solutions within the context of the business need. During this phase the business focus will be on defining the "to-be" processes, analyzing alternatives, analyzing risks, and ensuring security concerns are properly addressed and on-going efforts in this area are initiated.

3.3 OMB 300 Considerations[8]

The major emphasis at this stage is on establishing the value of the Information Technology (IT) investment and demonstrating that it has been thoroughly planned and will be properly managed. The agency will prepare a Business Case in the format required by OMB including Part 1 – Capital Asset Plan and Business Case[9] [and applies to all assets] and Part 2 – Additional Business Case Criteria for Information Technology. The OMB 300 Business Case justifies the costs associated with an IT investment throughout the life cycle of the investment). . The cost/spend plan represented by the summary table of OMB 300 and the acquisition plan applies to virtually all stages of the SDLC where work is done (and therefore money spent) but especially the first five phases of the SDLC (Initiation – System Concept Development – Planning – Requirements Analysis – Design) where the scope and nature of the work to be done is decided and projected costs are attached to units of work. The presentation of the Risk Management Plan is of particular importance to the first four phases of the SDLC (Initiation – System Concept Development – Planning – Requirements Analysis) where it may be a factor influencing go or no-go decisions.

The determination of performance measures and their actual measurement starts with the second phase of the SDLC (Systems Concept Development) and extends through the Steady State phase. This activity relates Agency strategy to accomplish "something" with how this project specifically will accomplish the "something".

[8]OMB 300 is Part 7 (section 300) of the Federal Office of Management and Budget's Circular No. A.11 "Planning, Budgeting, Acquisition, And Management Of Capital Assets." See *http://www.cio.gov/documents/s300.pdf*

[9]This is a form required by OMB 300 and is available with that material.

Enterprise Architecture (EA), security, and to some extent data considerations are applicable through all phases of the SDLC, beginning with the Initiation phase. EA and security concerns are especially important in the early phases of the SDLC and during the Implementation phase when the system is made operational.

4.0 PLANNING PHASE

During this phase planning for the project is carried out in detail, and control documents relating to that effort are produced. A Work Breakdown Structure (WBS) document will be prepared defining the product to be developed or produced by hardware, software, support, and/or service element, and relating the work scope elements to each other and to the end product(s). A detailed project schedule will be created. As in previous phases, reviews will again verify that the project is in alignment with the approved Enterprise Architecture and apply and report Earned Value Management System (EVMS) metrics (as required). Also in this phase the Data Council will review and approve the data architecture of the proposed system.

During this phase the business focus will be on monitoring progress and ensuring the system (hardware and software) as defined is complete, accurate, and available to all who may need the information.

During this phase, a Project Risk Management Plan, a System Security Plan, System Acquisition Plan, and Communications Plan will be drafted and approved. Details of these documents are offered below.

The following *types* of documents and artifacts, or their equivalents, will be prepared as a baseline for project control as required by FDA policy for project size, complexity, and duration:

- **Project Management Plan**: Project purpose, business and project goals and objectives, cost, schedule and performance goals (for both quality and quantity), scope and expectations, roles and responsibilities, assumptions and constraints, quality management approach, project management approach, and ground rules for the project, and a summary of the project costs will be documented in the plan.[10]
- **Project Schedule**: Identifies the major activities and deliverables, resources assigned, and dates associated with these.
- **Work Breakdown Structure**: Displays and defines the product to be developed or produced by hardware, software, support, and/or service element, and relates the work scope elements to each other and to the end product(s). A standard work breakdown structure consists of phases, activities, tasks, and deliverables. The framework of the WBS defines all authorized project work. For work being done by a contractor the Project Manager may expand the WBS into a Contract Work Breakdown Structure (CWBS) (this typically extends the CWBS a minimum of one level below the negotiated external reporting level, setting up the framework for work scope definitions and assignments to the functional organizations responsible for performing the work. The extended CWBS includes the levels at which required reporting information is

[10]Note that performance goals are required to be addressed by OMB 300. Performance goals, per that directive, should be linked to the annual FDA performance plan, should discuss the FDA's mission and strategic goals, and performance measures should be provided.

summarized for submittal to the customer. One and only one CWBS exists for each contract).[11]

? **Project Risk Management Plan**: identifies the risks to the project (i.e. which could impact its successful completion on-time, on budget and with the required functionality) and indicating how they will be tracked and mitigated.[12]

? **System Acquisition Plan**: Identifies elements such as acquisition alternatives (in-house development, contract development, COTS or turn-key), development or acquisition cost, implementation cost, contract review procedures, deficiency contingencies, post-award orientation steps (for contracted products), risk identification and other factors – such as the contract vehicle type – which should be considered.

? **System Security Plan**: Assessing the sensitivity and criticality, security risk, and documenting performance measures. The System Security Plan may also include plans for disaster recovery, and contingency and incident responses as these are developed.

? **Supplier Management Plan**: Documents activities which delineate the tasks and services best suited to contractor/vendor/supplier-sourcing; establishes an effective and efficient source selection process to provide qualified resources to undertake the work; ensures contractor/vendor/supplier-sourcing commencement occurs quickly so that timely commitment to baseline deliverables is achieved; integrates contractor/vendor/supplier-sourcing control processes into the regular project planning and control activities of the Project Team; provides for inspection, review, correction and acceptance/rejection of the product; provides for alternate sources and back up plans in event the contractor/vendor/supplier defaults.

? **Communications Plan**: Offers strategic and tactical guidance for creating and deploying targeted communications to key stakeholders involved with the planning and implementation of the system.

? **Requirements Management Plan**: Defines how requirements will be recorded; how requirements will be modified; and how requirements will be reconciled for final delivery of the product. Specific objectives include identifying the scope of what is managed, identifying roles and responsibilities of those involved in this management process, describing the processes and procedures to be used during this management process, and identifying the tools to be used.

- **Software Configuration Management** Plan: Identifies and defines the over-all methods and procedures necessary to perform software configuration management for the stated project, including maintaining a listing of all documentation being prepared for the system or component part of the system containing software; preventing the addition, deletion, or change to software items entered into the developmental configuration without traceability, and the unauthorized change of a baseline specification; enabling the retrieval of change status of baseline and documents entered into the developmental configuration; and providing a listing of documentation issue status for systems and Computer Software Configuration Items.

[11]Note that the WBS can be structured in either of two ways. The first approach structures the WBS primarily from a deliverables perspective, in that the highest level (Level 1) entries represent the major deliverables that the project is committed to create. The second approach is from a life cycle perspective, in that the highest-level entries in the WBS correspond to the major phases of the life cycle.

[12]Note that this is very different from the Security Risk Assessment that deals with a potential breach of data integrity, system operability, etc.

For larger, more complex, and more critical projects the following *types* of documents and artifacts, or their equivalents, may be prepared:

- **System Interface Document (System Boundary Document)**: Describes the relationship between two components of a system in terms of data items and messages passed, protocols observed and timing and sequencing of events. This type of document is typically used where complex interfaces exist between software components being developed by different teams. There will be an ICD to each external interface where the system is receiving data from an external system or sending data to an external system.
- **System Quality Assurance Plan**: Details the understanding of the quality needs of the organization, customer, and end users, provides mechanisms for tracing the system requirements and the quality goals, provides for appropriate test and evaluation processes to be *applied in each phase*.
- **Software Quality Assurance Plan**: Details the test and evaluation processes to be applied in each phase, the roles and responsibilities of personnel, documenting the test environments, and the acceptance criteria to be met, may be drafted.
- **Hardware Configuration Management Plan**: Identifies and defines the over-all methods and procedures necessary to perform hardware configuration management for the stated project, including maintaining a listing of all hardware required for the system; preventing the modification to hardware items entered into the developmental configuration without traceability, and the unauthorized change of a baseline specification; enabling the retrieval of change status of baseline and documents entered into the developmental configuration; and providing a listing of documentation issue status for systems and Computer Hardware Configuration Items.

NOTE: Documents drafted in previous phases may be updated and reviewed in this phase as new information becomes available, or requirements change.

This phase concludes with the completion of a Project Review, as a part of the continuing Quality Assurance effort, in which the documents produced are reviewed and approved, or approved with qualification (i.e. formal amendments are agreed to, documented, and assigned, with the understanding that the amended documents will act as the baseline and control from this point forward).

4.1 OMB 300 Considerations

In this phase the OMB 300 Business Case may be revised to better demonstrate to Departmental management and OMB that the IT investment is well managed, that it represents a strong business case and justification for the IT investment, and that it has met other Administration priorities. As before, Alternatives Analysis, Return on Investment and Net Present Value must be addressed. Some projects may take more than one round of alternative and cost-benefit analysis to develop the best fit for FDA needs. The cost/spend plan represented by the summary table of OMB 300 and the acquisition plan will need to be updated. The presentation of the Risk Management Plan is of particular importance in this phase where it may be a factor influencing go or no-go decisions.

5.0 REQUIREMENTS ANALYSIS PHASE

This phase formally defines the detailed functional user requirements using high-level requirements identified in the Initiation, System Concept, and Planning phases. It also delineates the requirements in terms of data, system performance, security, and maintainability. The requirements are defined in this phase to a level of detail sufficient for systems design to proceed. They need to be measurable, testable, and relate to the business need or opportunity identified in the Initiation Phase. The requirements that will be used to determine acceptance of the system are captured in the Quality Assurance Plan (or for efforts which do not require that document, in an appropriate Test Plan). As in previous phases, reviews will again verify that the project is in alignment with the approved Enterprise Architecture and apply and report on the Earned Value Management System metrics (as required). As in the previous phase, the Data Council will review and approve the updated data architecture of the proposed system.

Several trade-off decisions such as the decision to use COTS software products as opposed to developing custom software, or the decision to use an incremental delivery versus a complete, one-time deployment may be made in this phase.

Security requirements are developed in conjunction with the definition of the system requirements within the constraints detailed in the System Security Plan. Note that any planned COTS components in the system must be evaluated with regard to security specifications.

Extensive use may be made in this phase of prototyping. Prototyping provides a partial model of the future system and is useful in two situations:

- To offer early exposure to a proposed user interface for user feedback and design; and
- To identify risk areas where technology may be immature; in this case the prototyping exercise assists risk reduction in key project areas, providing clearer direction for the design and development phases.

Note that prototyping may also be used during the design phase to test the implications (i.e. the results) of some aspects of the requirements and design as these are being developed.

The purposes of this phase are to:

- Further define and refine the functional and data requirements and document them in the Systems Requirement Specification.
- Complete the analysis of the business process functions to be supported, e.g., verify what information drives the business process, what information is generated, who generates it, where does the information go, and who processes it.
- Conduct and document analysis of current system.
- Use coding standards and naming conventions established by Enterprise Architecture.
- Use database standards and naming conventions established by Enterprise Architecture
- Prepare (if required) a Purchase Request/Proposal (described below).
- Develop the test and evaluation requirements that will be used to determine acceptable system performance.
- Verify that the project is in alignment with the approved Enterprise Architecture.

During this phase the business focus will continue to be on monitoring progress and ensuring the system (hardware and software) requirements as defined are complete, accurate, and available to all whom may need the information.

The following *types* of documents and artifacts, or their equivalents, will be prepared as a baseline for project control as required by FDA policy for project size, complexity, and duration:

- **Current Systems Analysis Document**: A brief description of the current system
- **System Requirements Specification** (user requirement specification, functional specification): A document describing the requirements of a computer system from the user's point of view. An SRS document specifies:
  - The required behavior of a system in terms of input data, required processing, output data, operational scenarios and interfaces and
  - The attributes of a system including performance, security, maintainability, reliability, auditability, availability, safety requirements, and design constraints
- **Logical Data Model**: A graphical representation of the information requirements or common business practices in use in an organization area (division, department, function), and includes entity-relationship diagrams, entity definitions, and definitions of the attributes that make up the entities.
- **Logical Model Dictionary Report**: Provides the complete description of the logical data model including regular entities, reference entities and properties.
- **Software Preliminary Design Document**: A preliminary definition and description of the operations, interfaces, performance, and quality assurance requirements of the software to be developed.
- **Database Preliminary Design Document**: A preliminary definition and description of the operations, interfaces, performance, and quality assurance requirements of the database portion of the software to be developed.
- **System Test Plan** (including a Security Test and Evaluation Plan; and load testing [if required]): A plan detailing the types of tests (unit, formal (or integration), independent verification and validation) to be carried out, the acceptance criteria, roles and responsibilities, resources (hardware and software environments), and other elements relevant to test planning and execution. This plan details the testing of the integrated software/hardware system. In the case of larger or more complex projects there may be separate hardware and software test plans, as well as a test plan for the integrated system. Note that this document will be drafted in this phase, but subject to revision and added details as software development progresses).

For larger, more complex, and more critical projects the following *types* of documents and artifacts, or their equivalents, *may* be prepared:

- **Purchase Request/Proposal** (if required): Provides for the coordination and proper approval of the software or development services; describes and documents the implications for current and future data processing commitments and communicates with those responsible for the budget as necessary to determine all ramifications; areas addressed include cost, relation to the budget, vendor, target platform, mission need and other relevant information.

- **Software Development Plan**: Describes the process for designing, implementing, documenting, and testing the final software product
- **Software Coding Standards** (Conventions): A set of defined and documented standard coding practices which help ensure the readability, uniformity, and maintainability of software code and ensure that the product is in accordance with the approved Enterprise Architecture requirements.
- **Database Naming Standards** (Conventions): A set of defined and documented naming standards (or conventions) which help ensure the readability, uniformity, and maintainability of data elements and ensure that the product is in accordance with the approved Enterprise Architecture requirements.

NOTE: Documents drafted in previous phases may be updated and reviewed in this phase as new information becomes available, or requirements change.

This phase concludes with the completion of a System Requirements Review and /or Completed Software Requirements Review (as a part of the continuing Quality Assurance effort); where no hardware modifications or acquisitions are required for the software product, only the latter will take place. Where new or modified hardware is required, a System Requirements Review is also required. During these reviews the documents produced during this phase are reviewed and approved, or approved with qualification (i.e. formal amendments are agreed to, documented, and assigned, with the understanding that the amended documents will act as the baseline and control from this point forward).

5.1 OMB 300 Considerations

The major emphasis in the Requirements Analysis phase (from the perspective of OMB 300 compliance) will be on Enterprise Architecture, Security and Privacy, Acquisition Strategy and the Project and Funding Plan. The cost/spend plan represented by the summary table of OMB 300 and the acquisition plan will need to be updated. The presentation of the Risk Management Plan is of particular importance in this phase where it may be a factor influencing go or no-go decisions.

The Earned Value Measurement activities start with the development of the WBS in the Requirements Analysis phase but are most applicable in the Development through Implementation phases. After a project achieves steady state, annual operational reviews and alternatives analysis should be conducted to determine if the system is still the best solution to accomplish the work.

6.0 DESIGN PHASE

During this phase, the system is designed to satisfy the functional requirements identified in the previous phase. Since problems in the design phase can be very expensive to solve in later stages of the software development, a variety of elements are considered in the design to mitigate risk. These include:

- Identifying potential risks and defining mitigating design features.
- Drafting the Security Risk Assessment Plan and Contingency Plan.
- Determining the development environment.
- Defining major subsystems and their inputs and outputs.

- Developing high-level technical architecture, process models, and data models[13]
- Developing detailed data and process models including system inputs and outputs.
- Allocating processes to resources.
- Preparing detailed logic specifications for each software module.

The Software Preliminary Design Document receives a rigorous review by Center/Organization technical and functional representatives to ensure that it satisfies the business requirements. It progresses through a series of Design Reviews involving the Center/Organization IT Director and Business Sponsor. Once these individuals approve the design, the final Software Preliminary Design Document serves as a basis for the detailed design for the system. For larger or more complex systems a Software Detailed Design Document for software (and, as required, for databases) will also be drafted, reviewed, and approved. For smaller or less complex systems the Software Preliminary Design Document may serve both functions. Note that the construction of executable prototypes is encouraged to evaluate technology to support the business process. Concurrent with the development of the system design, work will begin on drafting (as required) the software documentation (including the Operations and Maintenance Manual). As in previous phases, reviews will again verify that the project is in alignment with the approved Enterprise Architecture and apply and report on EVMS metrics (as required).

During this phase the business focus will continue to be on monitoring progress and ensuring the system (hardware and software) requirements as defined are complete, accurate, and available to all whom may need the information.

The following *types* of documents and artifacts, or their equivalents, will be prepared as a baseline for project control as required by FDA policy for project size, complexity, and duration:

- **Requirements Traceability Matrix**: A matrix identifying the requirement specification item (by ID Number), the associated requirements statement, the related software module, the test specification associated with this, the relevant test case number, the verification status, and any modifications allowable or implemented. This ensures that all requirements from the highest to the lowest level are identified, associated with a section in the software/database design, that they are designed/implemented in the final system and that each is associated with a test exercise.
- **Security Risk Assessment Plan**: This document identifies any security risks and indicates how they will be tracked and mitigated. It also includes an evaluation of alternative security measures.
- **Contingency Plans**: These identify alternatives to allow system functionality to be achieved should the system security be breached.
- **Software Unit Test Plan**: A document describing the unit testing process in terms of the features to be tested, pass/fail criteria and testing approach, resource requirements and schedules.
- **Software Test Scripts/Cases/Scenarios**: Each script/case/scenario is a single test instance in terms of input data, test procedure, test execution environment and expected outcome. (Test scripts/cases/scenarios also reference test objectives such as verifying

[13]In most cases a well-written Requirements Specification will be sufficient, however the use of data flow diagrams, class diagrams, entity-relationship diagrams and the like, may be considered.

compliance with a particular requirement or execution of a particular program path. Note that this testing will also exercise any interfaces and interactions with the target hardware to be employed).

- **Disaster Recovery Plan**: A "Business Recovery" manual for use to execute a recovery of business operations due to a natural or man-made disaster. This is meant to enable the organization to function under emergency operating conditions and sets procedures that will allow the organization to assess the damage to and implement the recovery of any of records that may be affected by an emergency or disaster.
- **Project Implementation Plan**: Identifies the tasks, primary responsibilities, deliverables, and completion dates associated with the transition to use of the new system; in addition, the plan will contain (where applicable), the hardware, software, Internet connectivity, project personnel, and budgetary expenditures required to implement the target technology and information system environment.

For larger, more complex, and more critical projects the following *types* of documents and artifacts, or their equivalents, *may* be prepared:

- **Software Detailed Design Document**: A detailed definition and description of the operations, interfaces, performance, and quality assurance requirements of the software to be developed.
- **Database Detailed Design Document**: A detailed definition and description of the operations, interfaces, performance, and quality assurance requirements of the database portion of the software to be developed.
- **Software Build Plan**: Identifies the incremental development milestones and build decomposition plan for the project. These milestones are individual builds of software, interface components and database elements that must be developed to satisfy the functional requirements specified in the project System Requirements Specification (the portion related to the software product). This plan will show the milestones, build functionality (subset of master requirements) and overall software application architecture and the means by which the architecture will be partitioned into various builds.
- **Software Training Plan**: Identifies the scope, curriculum, techniques, roles and responsibilities, schedule and deliverables related to the training of personnel (including users, operators and maintenance personnel) employing the system.

NOTE: Documents drafted in previous phases may be updated and reviewed in this phase as new information becomes available, or requirements change.

This phase concludes with (at a minimum) the completion of a Preliminary Design Review as a part of the continuing Quality Assurance effort. During this review the documents produced during this phase are reviewed and approved, or approved with qualification (i.e. formal amendments are agreed to, documented, and assigned, with the understanding that the amended documents will act as the baseline and control from this point forward).

For larger projects or projects of more complexity, three optional reviews may be invoked. A Critical Design Review follows the completion of the final version of the Preliminary Design Description(s) (depending on the revisions of these) and ensures that the customer approves that level of design before coding beings. (In detail, this review verifies that the system as designed and presented in documents produced during this phase meets the goals and

objectives agreed upon, or that modifications are formally agreed to, documented, and responsibilities assigned [with the understanding that the amended documents will act as the baseline and control from this point forward]). Design reviews of the software and database Detailed Design Description(s) (where these are called for) may be conducted to ensure that the customer approves that level of design before coding beings.

6.1 OMB 300 Considerations

The major emphasis in the Design phase (from the perspective of OMB 300 compliance) will be on updating the cost/spend plan represented by the summary table of OMB 300 and the Acquisition Plan.

7.0 DEVELOPMENT PHASE

Effective completion of the previous stages is a key factor in the success of the Development phase. The Development phase consists of:

- Establishing the development environment.
- Translating the detailed requirements and design into system components.
- Reviewing the development of the software product.
- Testing individual elements (units) for usability.
- Submitting, recording and properly adjudicating all problem reports and Software or System Change Requests.
- Preparing any required development review reports.
- Conduct a risk assessment of security controls (in accordance with the Security Risk Assessment Plan).
- Implement and document the security controls and architecture.
- Preparing for integration and testing of the IT system.
- Documenting the *as built* system.
- Preparing software user documents and training materials (if required).

Within this phase, the detailed specifications produced during the design phase are translated into hardware, communications, and executable software. If an incremental or cyclical approach is desired, build development can be structured to support this. Using the cyclic build development approach a'large' effort can be subdivided (i.e. requirements allocated) to a release and sent through the lifecycle. Modifications or enhancements can be grouped into a release and sent through the lifecycle as a separate element of the system.[14]

[14]Using this approach, requirements for a specific build are defined and reviewed with the customer. The architecture for the build is defined in the test plans. The Test Plan for this approach contains both unit and formal testing elements, and as such is expanded to contain: requirements, design, test plans, and system environment changes pertaining to a given build. In effect, in this approach a build represents a total life cycle in miniature. During the course of the current build, the functionality of the system will be incrementally enhanced in a controlled and planned manner. If requirements change, they will be addressed in subsequent builds. The build and test process is repeated incrementally throughout the life of the project. Individual software units are developed and tested by the developers and reviewed by peers to insure that each unit satisfies the requirements allotted to it. All units must comply with established project coding style and conventions. Incremental Build Validation Testing is important to verify that the system fulfills its allotted requirements and to identify design and performance issues, which could affect future builds. Incremental testing as each build is completed results in a more stable, better-tested system by the time the system is tested as a whole.

Software (where applicable) is unit tested in a systematic manner during this phase. User involvement during development is an important factor in ensuring that the system is being developed according to the agreed to requirements. During this phase training materials and user documentation may be developed as required. As in previous phases, reviews will again verify that the project is in alignment with the approved Enterprise Architecture and apply and report EVMS metrics (as required).

During this phase the business focus will continue to be on monitoring progress and ensuring the system (hardware and software) requirements as defined are complete, accurate, and available to all whom may need the information.

The following *types* of documents and artifacts, or their equivalents, will be prepared as a baseline for project control as required by FDA policy for project size, complexity, and duration:

- **Software application modules**: Encapsulated software components that provide certain functionality or service that can be used in conjunction with other components to build applications.
- **Database modules**: The data layer, or "backend" of relational databases, which implement the functions to connect, query, and modify a database (as required).
- **Build Report and Location of Code**: A document identifying the results of the build (actions required to achieve final build, if any exceptions are found), and the location of the code on the hardware system.
- **Defect Log**: An identification of all known software defects, their symptoms and status; work-arounds may also be identified.
- **Software Design Description** (as built): A final *as built* definition and description of the operations, interfaces, performance, and quality assurance requirements of the software to be developed.
- **Database Design Description** (as built): A final, *as built* definition and description of the operations, interfaces, performance, and quality assurance requirements of the database portion of the software to be developed.
- **Security Controls and Architecture**: This may be included as a section in the Design Descriptions (for the as built database and software) or as a separate document).
- **Software Maintenance Plan**: Details the processes for corrective fixes, upgrades, testing, quality measurements, projected budgeting (for maintenance, upgrade and migration costs), minimizing downtime and indirect costs.[15]

Build integration then takes place in a sub-phase after the previous round of build development. The build integration activity provides developers a chance to couple the previous builds to the current build and test the combined functionality. Tests performed during build integration are focused on exercising the various connections and communications between components developed in separate builds. This incremental build-a-little, test-a-little helps insure system stability and insure that testing time does not get eliminated or negatively impacted at the end of the project when the schedule may be starting to slip.

[15]The term “maintenance” is understood here as the modification of a software product, after delivery, to correct faults, to improve performance or other attributes, or to adapt the product to a changed environment.

For larger, more complex, and more critical projects the following *types* of documents and artifacts, or their equivalents, *may* be prepared:

- **Software Unit Test Report**: A document describing the results of the unit testing process in terms of the features tested, testing approach the pass/fail results (including a list of known defects), and recommendations.
- **Change Request Report**(s): A report of all change requests submitted and their disposition (ex. accepted and product modified; rejected and reason why; accepted for scheduled future release, etc.).
- **Software Development Review Report**: A document indicating analysis (may include constraints and assumptions), design (program structure), documentation status, and testing (approaches and results) of the application software.
- **Database Review Report**: A document indicating analysis (may include constraints and assumptions), design (program structure), documentation status, and testing (approaches and results) of the database.
- **Software User Documentation**: Documentation associated with the employment of the software (e.g. User Manuals, Operations and Maintenance Manuals, etc.) will continue in development (if begun in a previous phase) or will begin development.
- **Software Training Materials**: Training materials (seminars, presentations, workbooks, self-study tutorials, etc.), which will accompany the release.

NOTE: Documents drafted in previous phases may be updated and reviewed in this phase as new information becomes available, or requirements change.

This phase concludes with the completion of a Test Readiness Review as a part of the continuing Quality Assurance effort. This review is meant to verify that the software product is of a sufficient state of readiness that it may be integrated and formally tested by an assigned test group (i.e. other than Development personnel).

7.1 OMB 300 Considerations

The major emphasis in the Development and Implementation phases (from the perspective of OMB 300 compliance) will be on demonstrating that the IT investment and funding plans are being met.

8.0 INTEGRATION AND TEST PHASE

Subsystem integration, system, security, user acceptance testing, and Certification and Accreditation (C&A) (for major applications or general support systems, as required by OMB A-130) are conducted during the integration and test phase. The user, with those responsible for quality assurance, validates that the functional requirements, as defined in the functional requirements document, are satisfied by the developed or modified system.

In this phase the System Security Plan is finalized. The security controls of the system, documented in the final System Security Plan, are then assessed for effectiveness. Multiple levels of testing may be performed, including:

- Formal system testing at the development facility by the developers and may include end users.

- Independent Security Testing and Evaluation (ST&E) as a deployed system will be performed by a test team independent (budgetary and managerial) of the development team. Upon completion of ST&E, the system must receive Certification and Accreditation, as required by FDA policy.

Requirements are traced throughout testing; user testing is performed to ensure the system satisfies requirements, while Security Testing and Evaluation is performed and all documentation is reviewed and accepted prior to acceptance of the system.

During this phase the business focus will continue to be on monitoring progress and ensuring the system (hardware and software) requirements as defined are complete, accurate, and available to all whom may need the information.

The following *types* of documents and artifacts, or their equivalents, will be prepared as a baseline for project control as required by FDA policy for project size, complexity, and duration:

- **System Test Report**: A report detailing the types of independent security review and testing carried out on the integrated software product, the acceptance criteria, roles and responsibilities, resources (hardware and software environments), and other elements used in the review and test execution.
- **Independent Security Test and Evaluation Report**: A report containing the results of the security test and evaluation process with a specified degree of technical, managerial, and financial independence from the development organization.
- **Security Risk Mitigation Plan**: This document identifies the risks identified during the security testing and evaluation and indicates how they will be tracked and mitigated.
- **Version Release Package**: For all projects this will contain at a minimum:

  - Known Problems Report/Resolution Plan: An identification of any outstanding problem reports and a brief indication of their proposed disposition.
  - *Preliminary* Security Certification Package.
  - Updated Documentation as required.

For larger, more complex, and more critical projects the following *types* of documents, artifacts, and controls, or their equivalents, may be invoked, and the associated documents *may* be prepared:

- **Independent Verification and Validation** (software and documentation): A verification and validation process performed with a specified degree of technical, managerial, and financial *independence* from the development organization. (*Verification* is defined as confirmation by examination and provision of objective evidence that specified requirements have been fulfilled. *Validation* is defined as the confirmation by examination and provision of objective evidence that the particular requirements for a specific intended use are fulfilled. In brief, validation asks and answers the question, "Are we doing the right thing?" while verification asks and answers the question, "Are we doing the thing right?")
- **Independent Verification and Validation** (software and documentation) **Report**: A report of the results of the independent verification and validation processes performed (this will include both a summary of the process and a detailing of the results).

- **Final versions of System Documents**: Manuals (User Manuals, Operations and Maintenance Manuals, etc.), which may accompany the release.
- **Final versions of Training Plans/Materials**: Training materials (seminars, presentations, workbooks, self-study tutorials, etc.), which may accompany the release.
- **Functional Configuration Audit Reports**: Report regarding the functional audit, the purpose of which is to provide an independent evaluation of software products, verifying that its configuration items' actual functionality and performance is consistent with the requirement specifications. Specifically this audit is held prior to the software delivery to verify that all requirements specified in the Software Requirements Specification have been met.
- **Physical Configuration Audit Reports**: Report regarding a type of configuration management audit. The objective of a physical audit is to provide an independent evaluation of a software product configuration item to confirm that components in the built version map to their specifications. The audit verifies that the software performs all the functions described in its design documentation and is ready for delivery.
- **Site Preparation Plan**: This may include a Site Survey (evaluating the existing condition and objectives for the site, and may include collecting floor layouts and checklists related to current infrastructure); Site Evaluation (a comprehensive study of the site and evaluation of the current system conditions and components such as applications for integration into the existing network); Site Preparation (implementing each recommended change specific to the site in question). After Site Preparation is complete, the site is ready to receive the system hardware and software. (NOTE: for some projects not requiring extensive hardware or software modification or support this may be included in the Project Implementation Plan

NOTE: Documents drafted in previous phases may be updated and reviewed in this phase as new information becomes available, or requirements change.

This phase concludes with the completion of a Deployment Readiness Review as a part of the continuing Quality Assurance effort, and the Certification and Accreditation Package. During the Deployment Readiness Review the reviewers will verify that the contents of the Version Release Package are in good order. The C&A process verifies that the system is operating with adequate security and must be completed prior to going into full production. The Certification Package (minimally the System Security Plan, ST&E Report, and Plan of Action and Milestones) is forwarded to the Information Systems Security Officer (ISSO) for review. The ISSO then forwards the package to the Certifying Official who will create an Accreditation Decision Letter and an accreditation documentation package and forward it to the Designated Approving Authority (DAA). The DAA will then issue an accreditation statement (Full Accreditation, Interim Accreditation or Accreditation Disapproval). The final Accreditation Package will include the Accreditation Letter, Security Plan, and the information documenting the accreditation decision. Verification and completion of these artifacts is the signal that the product is ready for deployment.

9.0 IMPLEMENTATION PHASE

This phase is initiated after the system has been tested and accepted by the user, and the system has received Accreditation (i.e. approval to Security from management to permit the system to operate in a production environment). In this phase, the system is installed to support the intended business functions. System performance is compared to performance objectives

established during the planning phase. Implementation includes user notification, user training, installation of hardware, installation of software onto production computers, and integration of the system into daily work processes. As in previous phases, reviews will again verify that the project is in alignment with the approved Enterprise Architecture and apply and report EVMS metrics (as required).

Note that once the system is installed a final security test will be made: all security features of the in-place system will be configured, activated, and tested. When it is determined that all security features are correctly implemented, the system will be made ready for processing. The system must, however, be tested again when new technical controls are implemented.

This phase continues until the system is operating in production in accordance with the defined user requirements.

During this phase the business focus will be on monitoring the implementation to ensure it is complete and effective, and on monitoring the activities of the Change Control Board, which will provide disposition to any requested modifications to the operational system.

The following *types* of documents and artifacts, or their equivalents, will be prepared as a baseline for project control as required by FDA policy for project size, complexity, and duration:

- **Project Implementation Risk Assessment Plan/Report**: The Plan identifies the risk associated with the implementation exercise and risks and indicates how they will be tracked and mitigated; the Report summarizes the activities carried out in accordance with the Plan and provides the results. (Note that in projects that are of medium scope, the risks may be identified and addressed in the Project Implementation Plan, although a separate report of their mitigation is required).
- **Certification**: A recommendation from the Security authority to the Certifying Official responsible for the system.
- **Accreditation**: Authority to Operate or Interim Approval to Operate (pending adjustments and final accreditation).

For larger, more complex, and more critical projects the following *types* of documents and artifacts, or their equivalents, *may* be prepared:

- **System Transition Plan**: Identifies the software support resources required (facilities, hardware, software) as well as documentation, personnel and outstanding issues; documents operational scenarios during the transition period; and details transition planning, including release process, data migration, conversion issues, installation details, problem resolution, and the transition schedule.

NOTE: Documents drafted in previous phases may be updated and reviewed in this phase as new information becomes available, or requirements change.

Pilot projects offer immediate results; they are the experience of how a complete system will operate, and will influence the drafting of “formal” requirements in the next life-cycle phase. In general, pilots allow the results of a project to be shown to decision-makers as evidence of the system's immediate value and provide a tangible way of communicating the potential of the system to skeptics within the organization. Pilots are used for:

- Verifying estimates of costs and benefits.
- Offering a chance for the organization to see a similar system running in production, and to evaluate its products.
- Providing a demonstration of limited facilities on a small area, using a system that may not be part of the final production system, mainly for development and hands-on experience.
- Providing early visibility of the system to management and users.
- Reducing risks associated with project before final commitment to full production is made.

This phase concludes with the completion of an Implementation Review as a part of the continuing Quality Assurance effort. During this review the reviewers will verify that the system was delivered and made operational in good order and that all elements of the deployment were successful, or that corrective plans are documented and agreed to, and resources are assigned and actions are being undertaken to see that the system will be operational in an acceptable time-frame.

10.0 EVALUATE PHASES

10.1 Steady State Phase

The system operation is ongoing. The system is monitored for continued performance in accordance with user requirements. When modifications or changes are identified, the system may re-enter the planning phase. The purpose of this phase is to:

- Operate and maintain the system.
- Conduct periodic assessments of the system to ensure the functional requirements continue to be satisfied.
- Conduct periodic re-certification and re-accreditation (audits and risk assessments based on the system Security Risk Assessment).
- Determine when the system needs to be modernized, replaced, or retired.
- Continue to perform security risk mitigation.
- Continue to perform security monitoring of proposed and implemented changes to the system.
- Revise and update system and supporting documentation as required (to include System Security Plans, Contingency Plans, etc. as well as User and Operator Manuals).

During this phase the business focus will be on analyzing alternatives to the current, operational system for effectiveness and efficiency, and making recommendations regarding these alternatives.

10.2 Disposition Phase

Disposition activities ensure the orderly termination of the system and preserve the vital information about the system so that some or all of the information may be reactivated in the future if necessary. Particular emphasis is given to proper preservation of the data processed by the system, so that the data can be effectively migrated to another system or archived for potential future access in accordance with applicable records management regulations and

policies. Signatures should be required to verify that all dependent users and impacted systems are aware of disposition.

During this phase the business focus will once again be on illustrating strategic alignment, analyzing "as-is" processes (including any architectural impact), and measuring current ("as-is") program performance (which will provide a baseline against which to measure alternatives).

10.3 OMB 300 Considerations

The major emphasis in these final phases (from the perspective of OMB 300 compliance) is on performance goals and measures, the IT Investment and Funding Plan, and the issues of meeting customer needs and continued alignment with the Enterprise Architecture.

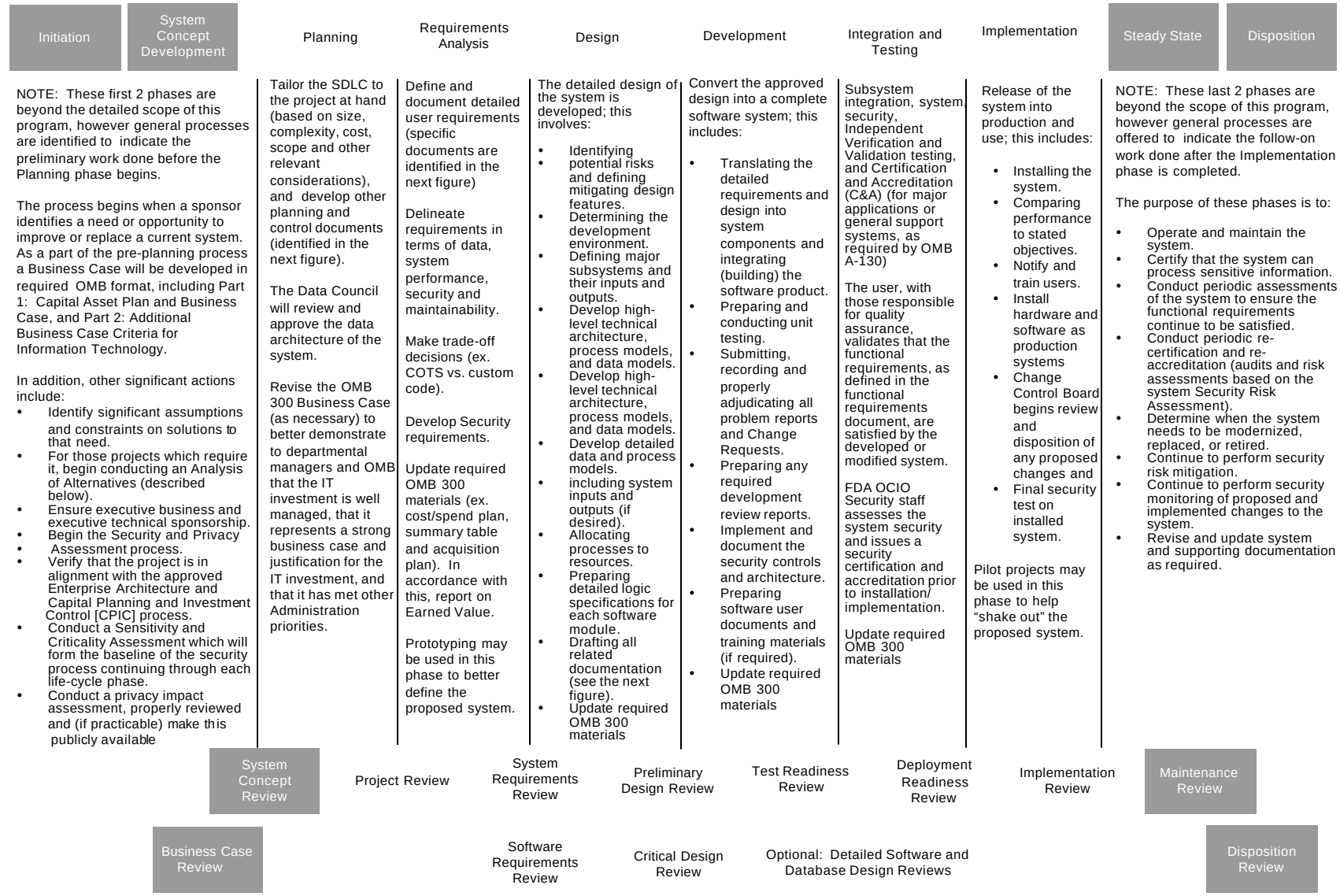
APPENDIX A: MATRIX OF SDLC DOCUMENT DEVELOPMENT

	Document Title C=Create, U=Update, F=Finalize	Planning	Requirements Analysis	Design	Development	Integration & Testing	Implementation
1	Accreditation Package						
2	Build Report and Location of Code				C	F	
3	Certification & Accreditation Package				C	U	F
4	Change Request Report				C	U	F
5	Communications Plan	C	U	U	U	F	
6	Contingency Plans			C	F		
7	Current System Analysis Document		C/F				
8	Database Design Description (as-built)				C/F		
9	Database Detailed Design			C	F		
10	Database Modules				C	F	
11	Database Naming Standards		C/F				
12	Database Preliminary Design Document		C/F				
13	Database Review Report				C/F		
14	Defect Log				C	F	
15	Disaster Recovery Plan			C	U	F	
16	Functional Configuration Audit Report					C/F	
17	Hardware Configuration Management Plan	C/F					
18	Independent Security Test & Evaluation Report					C/F	
19	Independent Verification & Validation					C/F	
20	Independent Verification & Validation Report					C/F	
21	Interface Control Document			C	F		
22	Logical Data Model		C/F				
23	Logical Model Dictionary Report		C/F				
24	Physical Configuration Audit Report					C/F	
25	Project Implementation Plan			C	U	F	
26	Project Implementation Risk Assessment Report						C/F
27	Project Management Plan	C	U	U	U	U	F
28	Project Schedule	C	U	U	U	U	U
29	Purchase Request/Request for Proposal		C/F				
30	Requirements Management Plan	C					
31	Requirements Traceability Matrix			C	U	U/F	
32	Risk Management Plan	C	U	U	U	U	U
33	Security Controls and Architecture				C/F		

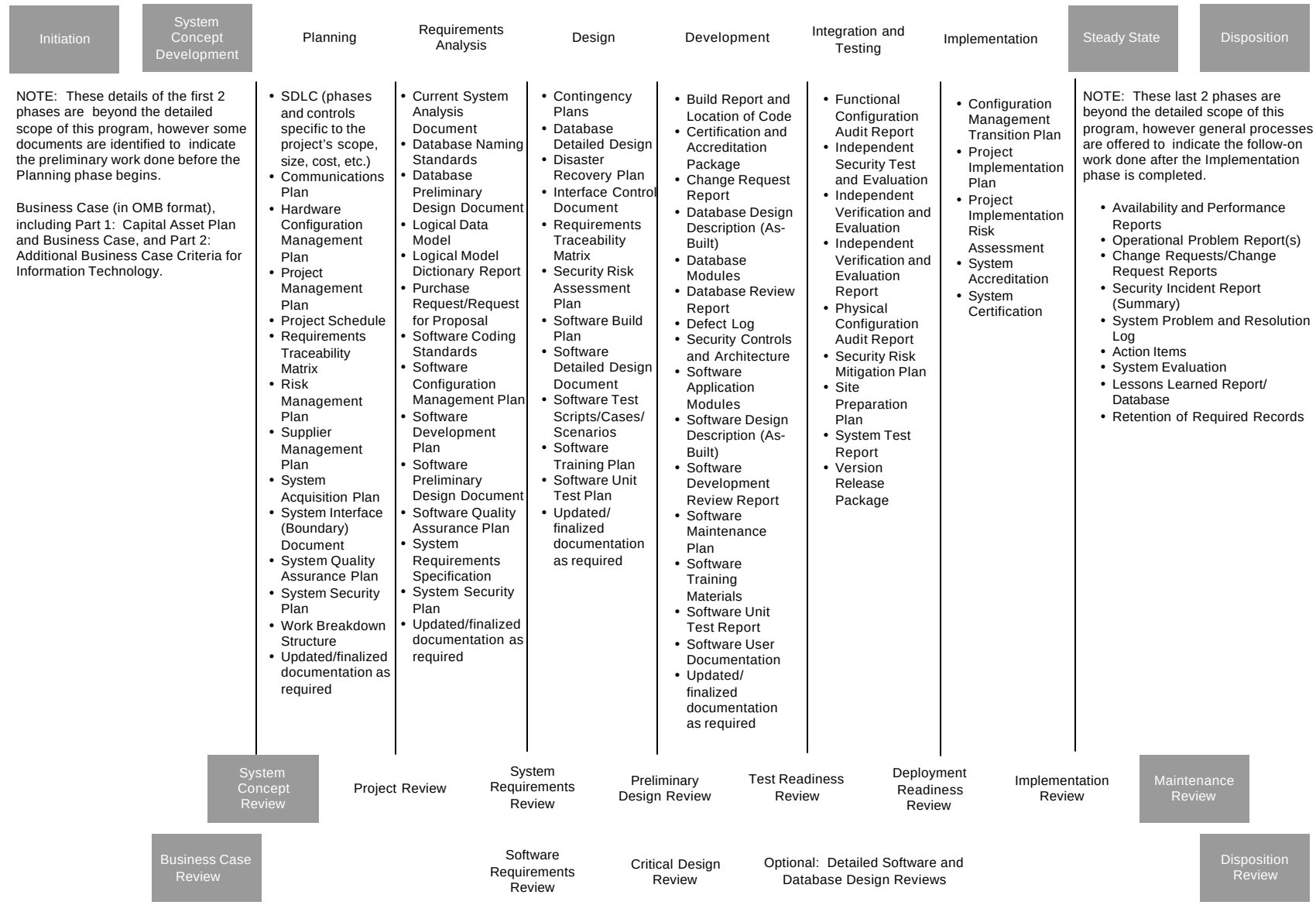
	Document Title C=Create, U=Update, F=Finalize	Planning	Requirements Analysis	Design	Development	Integration & Testing	Implementation
34	Security Risk Assessment Plan			C	U	F	
35	Security Risk Mitigation Plan					C/F	
36	Security Test and Evaluation Plan			C			
37	Security Test and Evaluation Report						
38	Site Preparation Plan					C/F	
39	Software Application Modules				C	F	
40	Software Build Plan			C	F		
41	Software Coding Standards		C/F				
42	Software Configuration Management Plan		C/F				
43	Software Design Description (as-built)				C/F		
44	Software Detailed Design Document			C/F			
45	Software Development Plan		C	F			
46	Software Development Review Report				C/F		
47	Software Maintenance Plan				C		F
48	Software Preliminary Design Document		C/F				
49	Software Quality Assurance Plan		C	U	U	F	
50	Software Test Plan (See System Test Plan)						
51	Software Test Scripts/Cases/Scenarios			C	U	F	
52	Software Training Materials				C	F	
53	Software Training Plan			C	F		
54	Software Unit Test Plan			C	F		
55	Software Unit Test Report				C/F		
56	Software User Documentation				C	F	
57	Supplier Management Plan	C/F					
58	System Accreditation						C/F
59	System Acquisition Plan	C					
60	System Interface (Boundary) Document	C/F					
61	System Certification						C/F
62	System Quality Assurance Plan	C	F				
63	System Requirements Specification		C/F				
64	System Security Plan	C	U	U	U	U	F
65	System Test Plan		C	U	U	F	
66	System Test Report					C/F	
67	System Transition Plan						C/F

68	Version Release Package				C/F
69	Work Breakdown Structure	C	U	F	

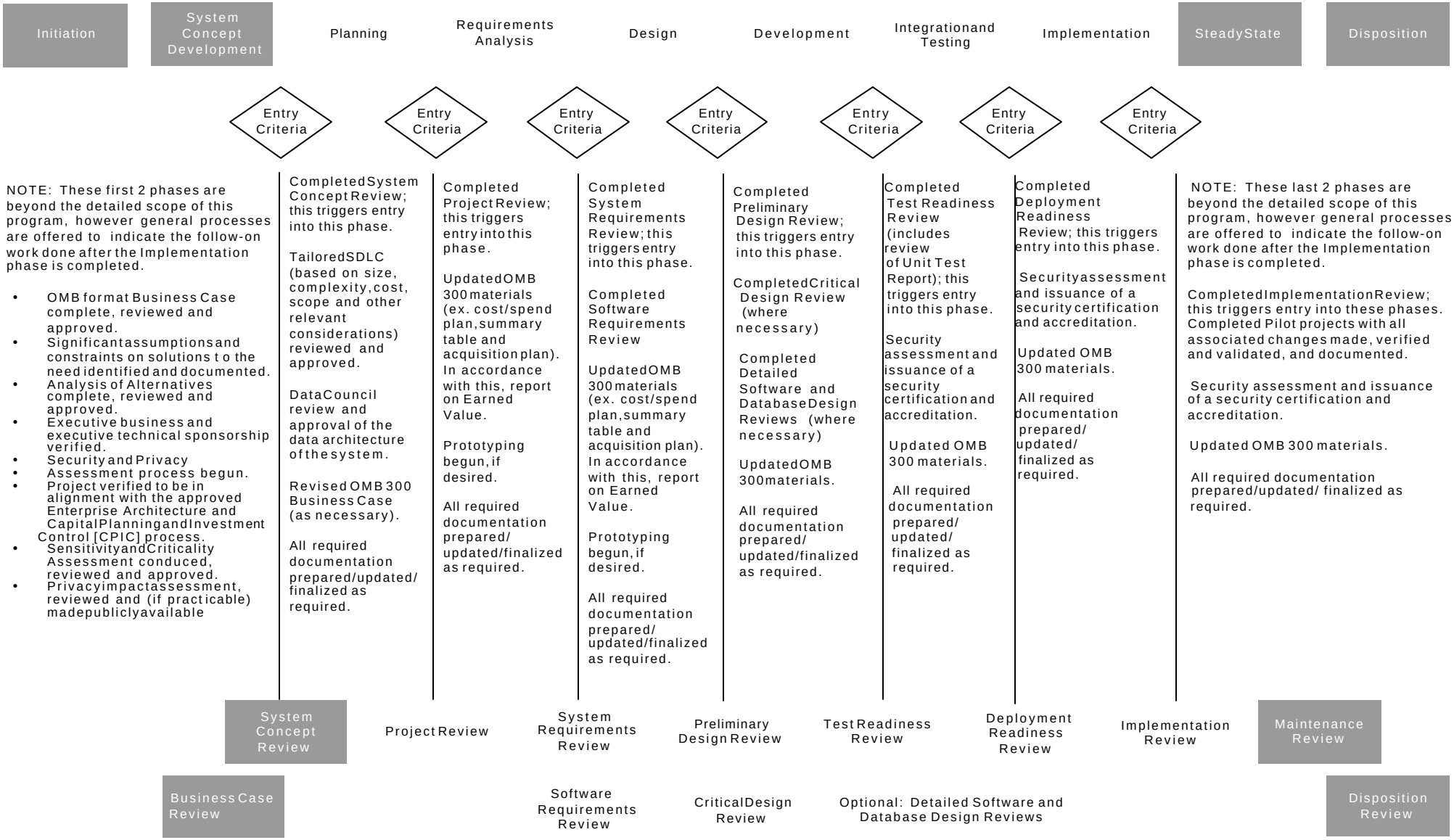
APPENDIX B: SDLC MAJOR PROCESS, DOCUMENTS, STAGE GATES AND COMMON CONTROLS OVERVIEW (S)



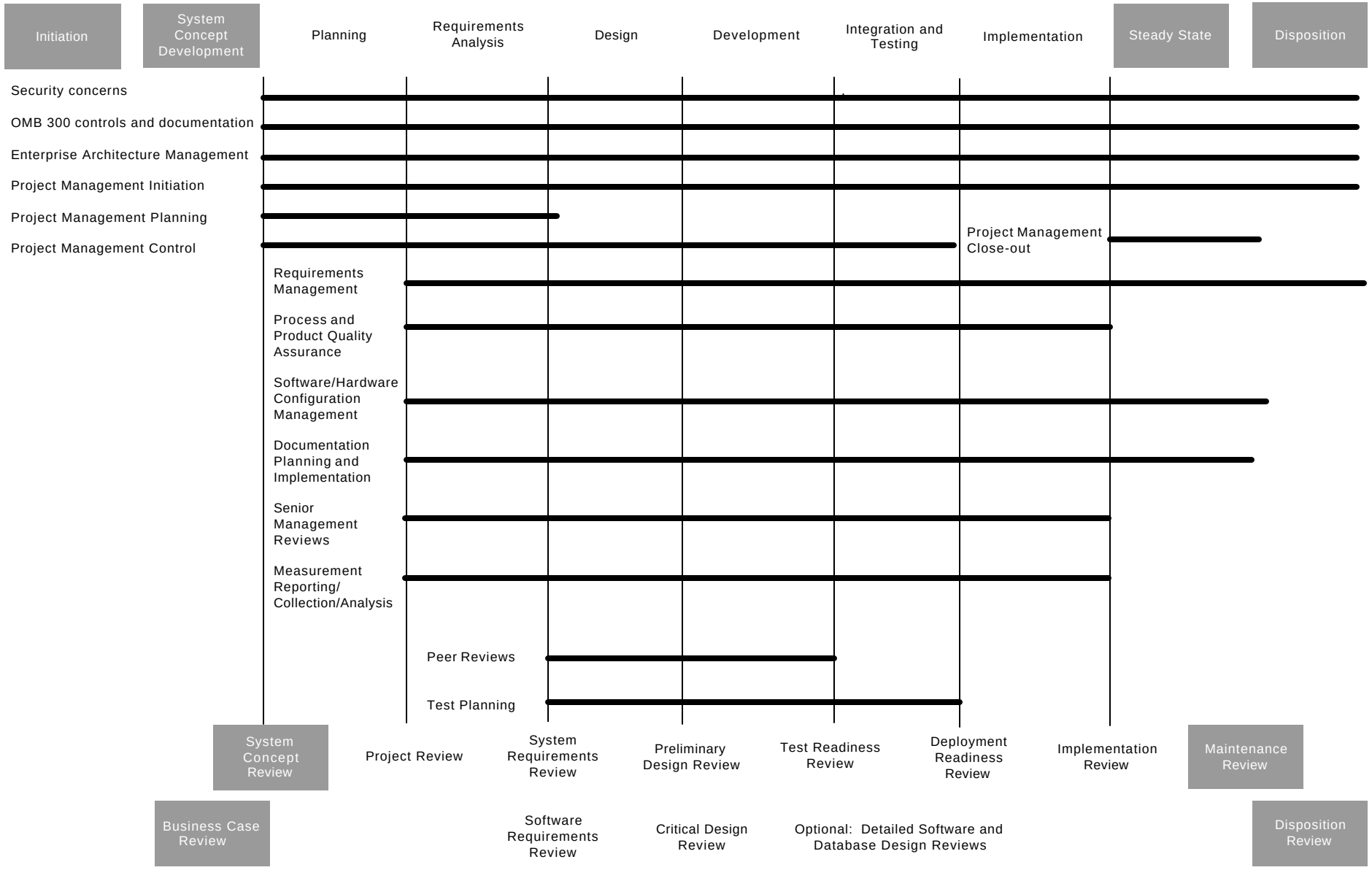
Major *Activities* Associated with Each SDLC Phase



Major Documentation Produced by SDLC Phase (Indicates Where Documents are First *Created*)



Stage Gates for Each SDLC Phase (Indicates Activities Which Must Be Complete For Each Phase to Begin)



Common Controls for SDLC Phases